Freeform Search

Database: US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: L4 and (NIC with table$ with address$)

Display: 10 **Documents in Display Format:** KWIC **Starting with Number** 1

Generate: ○ **Hit List** ◉ **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Wednesday, February 02, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=YES; OP=ADJ			
L9	L4 and (NIC with table$ with address$)	3	L9
L8	L4 and (NIC with map$ with address$)	0	L8
L7	L4 and (NIC with redundan$ with table$ with address$)	0	L7
L6	L4 and (NIC with redundan$ with map$ with address$)	0	L6
L5	L1 and ((network adj1 interface adj1 card) with redundan$)	8	L5
L4	L1 and (NIC with redundan$)	26	L4
L3	L2 and (redundan$.ab.)	1	L3
L2	L1 and (NIC or (network adj1 interface adj1 card)).ab.	92	L2
L1	714/$.ccls. or 709/$.ccls.	40216	L1

END OF SEARCH HISTORY

h e b b g e e e e e f ff e ch

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L9: Entry 2 of 3 File: USPT Aug 7, 2001

DOCUMENT-IDENTIFIER: US 6272113 B1
TITLE: Network controller system that uses multicast heartbeat packets

Detailed Description Text (6):
As described more fully below, each of the NICs 122 enables the computer system to communicate with other devices on a corresponding network. The computer system 100 may be coupled to at least as many networks as there are NICs 122, or two or more of the NICs 122 may be coupled to the same network via a common network device, such as a hub or a switch. When multiple NICs 122 are coupled to the same network, each provides a separate and redundant link to that same network for purposes of fault tolerance or load balancing, otherwise referred to as load sharing. Each of the NICs 122, or N1 -N4, preferably communicate using packets, such as Ethernet.TM. packets or the like. As known to those skilled in the art, a destination and source address is included near the beginning of each Ethernet.TM. packet, where each address is at least 48 bits for a corresponding media access control (MAC) address. A directed or unicast packet includes a specific destination address rather than a multicast or broadcast destination. A broadcast bit is set for broadcast packets, where the destination address are all ones (1's). A multicast bit in the destination address is set for multicast packets.

Detailed Description Text (17):
FIG. 5 is a block diagram illustrating one embodiment in which the intermediate driver 310 defines a Heartbeat Multicast Address (HMC) and where the intermediate driver 310 causes each NIC team member to register the HMC address. Upon power-up, boot or initialization, the O/S 301 starts each of the NIC drivers D1-D4 and the intermediate driver 310. The intermediate driver 310 detects and collects any and all multicast addresses (not shown) supported by each supported higher level protocol, such as the TCP/IP 302, IPX 304 and NetBEUI 306, and appends its own multicast address(es), which includes the HMC address. The intermediate driver 310 then requests that each NIC driver D1-D4 register the list of multicast addresses, including the HMC address. As shown in FIG. 5, each NIC driver D1-D4 and the corresponding NICs N1-N4 are programmed to detect the single node address A and the HMC address. It is noted that although only the HMC address is shown, each NIC driver D1-D4 may be programmed with a table of multicast addresses. The intermediate driver 310 also includes heartbeat logic 502 that includes memory for storing the HMC address and a status table 504 that maintains the status of each of the ports P1-P4 (including the NIC drivers D1-D4 and the NICs N1-N4) of the team. The intermediate driver 310 also includes a timer or timer logic 506 that determines the heartbeat period for checking the status of the ports P1-P4. The heartbeat period is referred to as the HEARTBEAT_TIMER_SPEED.

Current US Cross Reference Classification (3):
714/712

Previous Doc Next Doc Go to Doc#

h e b b g e e e f e e e g

L5	L1 and ((network adj1 interface adj1 card) with redundan$)	8	L5
L4	L1 and (NIC with redundan$)	26	L4
L3	L2 and (redundan$.ab.)	1	L3
L2	L1 and (NIC or (network adj1 interface adj1 card)).ab.	92	L2
L1	714/$.ccls. or 709/$.ccls.	40216	L1

END OF SEARCH HISTORY

Refine Search

Search Results -

Term	Documents
NETWORK	288848
NETWORKS	114293
INTERFACE	426513
INTERFACES	122483
CARD	121834
CARDS	66411
REDUNDAN$	0
REDUNDAN	5
REDUNDANAT	1
REDUNDANC	1
REDUNDANCCY	1
(L1 AND ((NETWORK ADJ1 INTERFACE ADJ1 CARD) WITH REDUNDAN$)).USPT.	8

There are more results than shown above. Click here to view the entire set.

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L5

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Wednesday, February 02, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=USPT; PLUR=YES; OP=ADJ		

h e b b cg b e e ch

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L3: Entry 1 of 1 File: USPT Jul 15, 2003

DOCUMENT-IDENTIFIER: US 6594776 B1
TITLE: Mechanism to clear MAC address from Ethernet switch address table to enable network link fail-over across two network segments

Abstract Text (1):
There is provided a communication network and method for enhancing server availability to client PCS which includes two Ethernet switches. Each one of the two Ethernet switches is connected to a corresponding one of the primary and secondary network interface cards in the file server PC. The two Ethernet switches are interconnected together through an uplink port. As a result, redundancy has been effectively and efficiently provided against the failure of either one of the two switches in order to enable link fail-over across two network segments.

Current US Original Classification (1):
714/4

Current US Cross Reference Classification (2):
714/11

Previous Doc Next Doc Go to Doc#

h e b b g e e e f e e e g

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L9: Entry 1 of 3 File: USPT Jul 15, 2003

DOCUMENT-IDENTIFIER: US 6594776 B1
TITLE: Mechanism to clear MAC address from Ethernet switch address table to enable network link fail-over across two network segments

Brief Summary Text (6):
A primary link or cable 18 has its one end 20 coupled to a first or primary network interface card (NIC) mounted within the file server PC 14 and its other end 22 connected to the Ethernet switch 16. In order to enhance network link availability, a secondary link or cable 24 is provided which has its one end 26 coupled to a second or redundant network interface card (NIC) in the file server PC and its other end 28 connected to the Ethernet switch 16. Under normal operation, the file server PC 14 uses the primary NIC and the primary link 18 to transmit and receive traffic. When the primary NIC fails and is unable to transfer data packets over the primary link 18 and the switch 16 to the client PCS, the redundant NIC will take over and is placed into service for performing the data transfer over the secondary link 24 and the switch 16. This operation is seamless and does not effect the normal network operation. When the primary NIC comes back on, the traffic is automatically switched from the secondary NIC/link 24 to the primary NIC/link 18.

Detailed Description Text (2):
Referring now in detail to the drawings, there is shown in FIG. 2 a graphical representation of a net-worked client-server computer system 110 for enhancing server availability to client PCS, constructed in accordance with the principles of the present invention. The client-server computer system 110 includes a plurality of client personal computers (PCS) 112a-112e which communicate with a file server PC 114 so as to access shared resources. The server PC 114 has mounted therein first (primary) and second (default or redundant) network interface cards (NIC) configured as a fail-over pair. Both the primary NIC and the secondary NIC are programmed with a single, common media access control (MAC) address which is used to identify the server PC 114.

Detailed Description Text (4):
In the normal operating condition, the Ethernet switch A (124) has initially stored in its address table the MAC address of the primary NIC. Therefore, the client PC 112a, for example, is able to be connected to the file server PC 114. In the event that the primary link 118 connected to the primary NIC should malfunction or fail, such as the cable or link being disconnected and/or the switch 124 failing, the secondary NIC will then take over. Further, the switch A (124) will delete the MAC address of the server PC 114 from its address table. As a result, the secondary NIC will send an LLC broadcast packet to the switch B (132).

Detailed Description Text (6):
Now if the primary link or cable 118 is re-connected or reinstated and/or the primary switch 124 is restored, the server PC software will switch control from the secondary NIC to the primary NIC. However, due to the fact that the secondary NIC remains connected to the backup Ethernet switch B (132) through the secondary link or cable 128, the backup switch B would still continue to have the MAC address of the server PC 114 stored in its address table. As a result, if the client PC 112a were connected to the backup switch B (132) it would not be able to be connected to

the server PC 114 due to the fail-back process of the primary NIC. In view of this, when the server PC 114 completes the fail-back, the secondary NIC is also used to break the secondary link or cable 128 for a short period of time. In this fashion, the backup switch B will cause the MAC address of the server PC 114 to be deleted from its address table. As a consequence, the client server 112a will again be able to be connected to the server PC 114 via the primary switch A and the primary link 118.

Detailed Description Text (8):
In block 304, the driver monitors periodically the status of the primary link 118 in order to determine whether there is a failure in the primary NIC, primary link, or switch A. If the answer is "NO", then the process goes to the block 306 where the primary NIC is continued to be used for sending and/or receiving of the frames and is looped back to the block 304. If the answer is "YES" from the block 304, the process will proceed to block 308 in which the driver initiates a fail-over process by transferring control to the secondary NIC. Upon finding that there is a failure in the primary link 118 to the primary NIC, the primary switch A in the block 310 will remove the NIC's address from its address table.

Detailed Description Text (9):
Once the fail-over process has been completed and the secondary NIC is ready to take over the network traffic from the primary NIC, the driver in block 312 will send a broadcast LLC frame using the secondary NIC to the secondary link 128 and the switch B. In the block 314, when the switch B receives this LLC frame, it will add the MAC address to its address table. In block 316, the driver will continue to monitor periodically the status of the primary link 118 in order to determine whether the primary NIC is back on-line. If the answer is "NO", then the process goes to the block 318 where the secondary NIC is continued to be used for sending and/or receiving of the frames and is looped back to the block 316. If the answer is "YES" from the block 316, the process will proceed to block 322 in which the driver initiates a fail-back process by transferring control to the primary NIC.

Detailed Description Text (10):
Upon finding that the failure has been repaired, such as re-connecting of a disconnected cable or replacing the failed NIC with a new one (i.e., "Hot Swap" procedure), in the block 322 the link pulses being transmitted from the secondary NIC are then turned off for a short period of time which is accomplished by resetting a device in the physical layer. Since the device in the physical layer requires a certain amount of time before re-initialization, the link pulses will be turned off during this time interval. This causes the secondary switch B to assume that the secondary link 128 has failed. As a result, the secondary switch B in block 324 will remove the NIC's address from its address table.

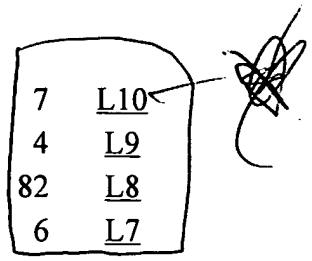
Detailed Description Text (11):
Thereafter, the driver in block 326 will again send a broadcast LLC frame using the primary NIC to the primary link 118 and the switch A. In the block 328, when the switch A receives this LLC frame, it will add the MAC address to its address table again. Since the LLC (Logical Link Control) frame is broadcasted, the switch A (124) will forward the LLC frame to the switch B (130) via the uplink port 134. This causes the switch B to associate the server PC's MAC address with the uplink port 134 in its address table. As a result, if a client PC should be connected to the switch B, then such client PC would be caused to be connected to the server PC 114 via the uplink port 134, the primary switch A, and the primary link 118. The fail-back process is completed in the End block 330. However, the overall process is looped back to the block 304 as indicated by the line 332 in order to repeat the same.

Current US Original Classification (1):
714/4

Current US Cross Reference Classification (2):
714/11

Previous Doc Next Doc Go to Doc#

h e b b g e e e f e e e g



L10	(L8 or L7) and ((address adj1 table$) or (address adj1 map$))	7	L10
L9	(L8 or L7) and ((address adj1 table$) or (adress adj1 map$))	4	L9
L8	(NIC or (network adj1 interface)) with (redundancy)	82	L8
L7	L5 and (redundancy).ab.	6	L7
L6	L5 and redundancy	246	L6
L5	(NIC or (network adj1 interface)).ab.	1447	L5
L4	L3 and redundancy	2	L4
L3	L1 and (dynamic$ with map$ with virtual with address$)	5	L3
L2	L1 and (dynamic$ with map$ with virtual with address$).ab.	0	L2
L1	(network$ and communicat$).ab.	16476	L1

END OF SEARCH HISTORY

h e b b cg b e e ch

US006594776B1

(12) **United States Patent**
Karighattam et al.

(10) **Patent No.: US 6,594,776 B1**
(45) **Date of Patent: Jul. 15, 2003**

(54) **MECHANISM TO CLEAR MAC ADDRESS FROM ETHERNET SWITCH ADDRESS TABLE TO ENABLE NETWORK LINK FAIL-OVER ACROSS TWO NETWORK SEGMENTS**

(75) Inventors: **Kishore Karighattam**, Sunnyvale, CA (US); **Sujalendu Das**, San Jose, CA (US)

(73) Assignee: **Advanced Micro Devices, Inc.**, Sunnyvale, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 482 days.

(21) Appl. No.: **09/604,818**

(22) Filed: **Jun. 28, 2000**

(51) **Int. Cl.**[7] **H02H 3/05**
(52) **U.S. Cl.** **714/4**; 370/386; 714/11
(58) **Field of Search** 714/4, 11; 709/200, 709/223, 225, 227, 239; 370/469, 386, 395

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,781,530 A * 7/1998 Segal 370/220
5,978,373 A * 11/1999 Hoff et al. 370/392
6,188,689 B1 * 2/2001 Katsube et al. 370/389
6,226,677 B1 * 5/2001 Slemmer 709/227
6,397,345 B1 * 5/2002 Edmonds et al. 714/4
6,412,079 B1 * 6/2002 Edmonds et al. 714/11

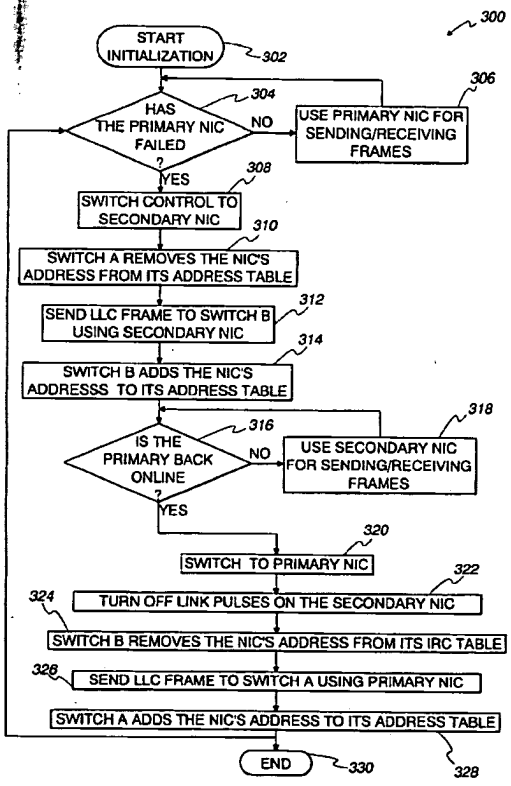
* cited by examiner

Primary Examiner—Dieu-Minh Le
(74) *Attorney, Agent, or Firm*—Davis Chin

(57) **ABSTRACT**

There is provided a communication network and method for enhancing server availability to client PCS which includes two Ethernet switches. Each one of the two Ethernet switches is connected to a corresponding one of the primary and secondary network interface cards in the file server PC. The two Ethernet switches are interconnected together through an uplink port. As a result, redundancy has been effectively and efficiently provided against the failure of either one of the two switches in order to enable link fail-over across two network segments.

2 Claims, 3 Drawing Sheets



L3	L1 and (dynamic$ with map$ with virtual with address$)	5	L3
L2	L1 and (dynamic$ with map$ with virtual with address$).ab.	0	L2
L1	(network$ and communicat$).ab.	16476	L1

END OF SEARCH HISTORY

h e b b cg b e e ch

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L3: Entry 1 of 5 File: USPT Oct 5, 2004

DOCUMENT-IDENTIFIER: US 6801949 B1
TITLE: Distributed server cluster with graphical user interface

Abstract Text (1):
A scalable, distributed, highly available, load balancing server system having multiple machines is provided that functions as a front server layer between a network (such as the Internet) and a back-end server layer having multiple machines functioning as Web file servers, FTP servers, or other application servers. The front layer machines comprise a server cluster that performs fail-over and dynamic load balancing for both server layers. The operation of the servers on both layers is monitored, and when a server failure at either layer is detected, the system automatically shifts network traffic from the failed machine to one or more operational machines, reconfiguring front-layer servers as needed without interrupting operation of the server system. The server system automatically accommodates additional machines in the server cluster, without service interruption. The system operates with a dynamic reconfiguration protocol that permits reassignment of network addresses to the front layer machines. The front layer machines perform their operations without breaking network communications between clients and servers, and without rebooting of computers.

Detailed Description Text (98):
As noted above, a variety of unique server functional features are provided by a server cluster constructed and operated in accordance with the invention. The server cluster functions as a gateway and dynamically maps virtual network addresses, which are the network addresses available to nodes outside the gateway, to primary network addresses, which correspond to the MAC hardware addresses of the gateway nodes. With application software in accordance with the invention, the machines of the gateway server cluster communicate with the network through the virtual addresses of the front-layer subnet and communicate with one or more subnets of a back-layer group of nodes, as depicted in FIG. 3. The unique server cluster functional features will be described next.

Previous Doc Next Doc Go to Doc#

US006801949B1

(12) **United States Patent**
Bruck et al.

(10) **Patent No.: US 6,801,949 B1**
(45) **Date of Patent: Oct. 5, 2004**

(54) **DISTRIBUTED SERVER CLUSTER WITH GRAPHICAL USER INTERFACE**

(75) Inventors: **Jehoshua Bruck**, La Canada, CA (US); **Vasken Bohossian**, Pasadena, CA (US); **Chenggong Charles Fan**, Fremont, CA (US); **Paul LeMahieu**, Pasadena, CA (US); **Philip Love**, Pasadena, CA (US)

(73) Assignee: **Rainfinity, Inc.**, Mountain View, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/566,592**

(22) Filed: **May 8, 2000**

Related U.S. Application Data

(63) Continuation of application No. 09/548,188, filed on Apr. 12, 2000, which is a continuation of application No. 09/437,637, filed on Nov. 10, 1999.
(60) Provisional application No. 60/128,872, filed on Apr. 12, 1999.

(51) **Int. Cl.**[7] **G06F 15/16**
(52) **U.S. Cl.** **709/232**; 709/234; 709/238
(58) **Field of Search** 709/232, 234, 709/235, 238–242, 220, 221, 102, 103, 105; 370/229

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,792,941	A	12/1988	Yanosy, Jr. et al.	370/58
5,191,651	A *	3/1993	Halim et al.	709/250
5,341,477	A *	8/1994	Pitkin et al.	709/226
5,530,897	A	6/1996	Meritt	395/829
5,550,816	A	8/1996	Hardwick et al.	370/60
5,636,216	A	6/1997	Fox et al.	370/402
5,729,681	A	3/1998	Aditya et al.	395/200.1
5,774,660	A *	6/1998	Brendel et al.	709/201
5,774,668	A	6/1998	Choquier et al.	395/200.53
5,790,804	A	8/1998	Osborne	709/105
5,825,772	A	10/1998	Dobbins et al.	370/396

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

WO	9826559	6/1998
WO	WO 99/17217	4/1999
WO	9933227	7/1999
WO	WO 00/62502	10/2000
WO	WO 01/35601	5/2001

OTHER PUBLICATIONS

Rainfinity Inc., Rainfinity unveils Rainwall—Industry's first fully distributed clustering solution for Internet gateways, Press release Apr. 14, 1999.*

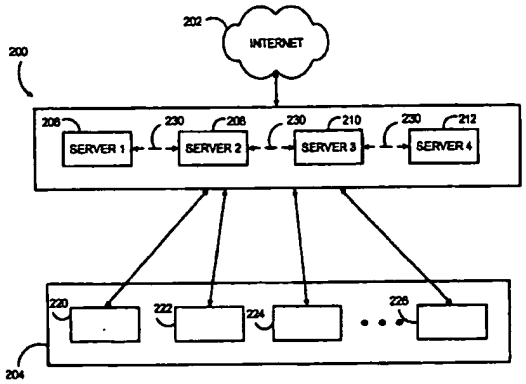
(List continued on next page.)

Primary Examiner—Frantz B. Jean
(74) *Attorney, Agent, or Firm*—Heller Ehrman White & McAuliffe

(57) **ABSTRACT**

A scalable, distributed, highly available, load balancing server system having multiple machines is provided that functions as a front server layer between a network (such as the Internet) and a back-end server layer having multiple machines functioning as Web file servers, FTP servers, or other application servers. The front layer machines comprise a server cluster that performs fail-over and dynamic load balancing for both server layers. The operation of the servers on both layers is monitored, and when a server failure at either layer is detected, the system automatically shifts network traffic from the failed machine to one or more operational machines, reconfiguring front-layer servers as needed without interrupting operation of the server system. The server system automatically accommodates additional machines in the server cluster, without service interruption. The system operates with a dynamic reconfiguration protocol that permits reassignment of network addresses to the front layer machines. The front layer machines perform their operations without breaking network communications between clients and servers, and without rebooting of computers.

8 Claims, 38 Drawing Sheets



Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L3: Entry 2 of 5 File: USPT Feb 3, 2004

DOCUMENT-IDENTIFIER: US 6687735 B1
TITLE: Method and apparatus for balancing distributed applications

Abstract Text (1):
An improved method and apparatus for balancing distributed applications within a client/server network, such as a cable television network, is disclosed. In one aspect of the invention, a method of balancing the load of distributed application client portions (DACPs) across various server portions (DASPs) and server machines is disclosed. Statistics are maintained by one or more software processes with respect to the available resources of the servers and their loading; new process threads and/or distributed application server portions are allocated across the servers to maintain optimal system performance as client device loading increases or changes. In another aspect of the invention, a novel object-oriented distributed application software architecture employing both vertical and horizontal partitions and "mutable" (i.e., transportable) objects is disclosed. The mutable objects may reside on either the server or client portions of the distributed application while maintaining at least one network partition. A runtime environment adapted for the operation of the foregoing object-oriented distributed application, including an efficient message protocol useful for interprocess communication, is also disclosed. Methods for downloading the DACP from the servers, and scaling the DACP at download based on client device configuration, are further disclosed.

Detailed Description Text (72):
The message protocol (MP) of the invention further assigns virtual addresses (VAs) to DASPs and DACPs, so that distributed application portions can move dynamically within the distributed application balancing system network. Servers associated with the distributed application balancing system network have records, for example, in their respective distributed application balancing system databases 706 that contain this dynamic mapping of virtual addresses. Clients on the network are only given those virtual addresses necessary to their communications needs. In one embodiment, however, clients can discover the virtual address of other DASPs and DACPs by sending a query message to the server farm 708. Discovery of such virtual addresses may be performed for, inter alia, identifying a well known server that provides a specific service, or to find applications of the same type running on other client devices.

Previous Doc Next Doc Go to Doc#

US006687735B1

(12) **United States Patent**
Logston et al.

(10) **Patent No.: US 6,687,735 B1**
(45) **Date of Patent: Feb. 3, 2004**

(54) **METHOD AND APPARATUS FOR BALANCING DISTRIBUTED APPLICATIONS**

(75) Inventors: **Gary Logston**, Poway, CA (US); **Patrick Ladd**, San Marcos, CA (US)

(73) Assignee: **Tranceive Technologies, Inc.**, Carlsbad, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 518 days.

(21) Appl. No.: **09/583,064**

(22) Filed: **May 30, 2000**

(51) **Int. Cl.**[7] **G06F 15/16**

(52) **U.S. Cl.** **709/203**; 709/217; 709/219; 709/220; 709/316; 709/328; 709/329; 370/486; 370/487

(58) **Field of Search** 709/203, 217, 709/219, 220, 316, 328, 329; 370/486, 487

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,969,092	A		11/1990	Shorter
4,991,089	A		2/1991	Shorter
5,457,797	A		10/1995	Butterworth et al.
5,555,244	A		9/1996	Gupta et al.
RE35,448	E		2/1997	Shorter
5,606,493	A		2/1997	Duscher et al.
5,664,093	A		9/1997	Barnett et al.
5,673,265	A		9/1997	Gupta et al.
5,740,176	A		4/1998	Gupta et al.
5,799,017	A		8/1998	Gupta et al.
5,818,448	A		10/1998	Katiyar
5,826,085	A	*	10/1998	Bennett et al. 709/227
5,864,542	A		1/1999	Gupta et al.
5,898,839	A	*	4/1999	Berteau 709/202
5,915,090	A	*	6/1999	Joseph et al. 709/203
5,920,868	A		7/1999	Fowlow et al.
5,924,094	A		7/1999	Sutter
5,958,009	A		9/1999	Friedrich et al.
5,958,012	A		9/1999	Battat et al.
5,966,531	A		10/1999	Skeen et al.
5,993,038	A		11/1999	Sitbon et al.
6,026,404	A		2/2000	Adunuthula et al.
6,028,846	A		2/2000	Cain
6,047,323	A		4/2000	Krause
6,055,537	A		4/2000	LeTourneau
6,058,106	A		5/2000	Cudak et al.
6,067,545	A		5/2000	Wolff
6,067,577	A		5/2000	Beard
6,073,163	A	*	6/2000	Clark et al. 709/229

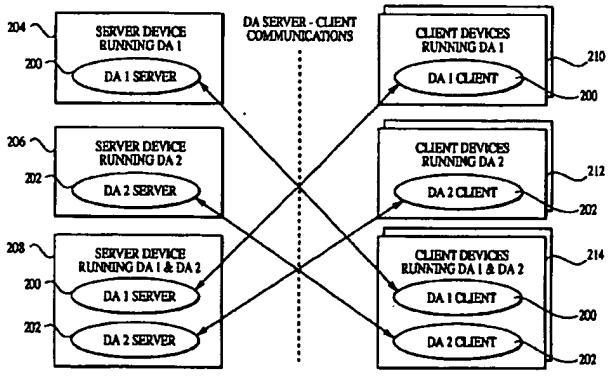
(List continued on next page.)

Primary Examiner—Zarni Maung
Assistant Examiner—Jinsong Hu
(74) *Attorney, Agent, or Firm*—Gazdzinski & Associates

(57) **ABSTRACT**

An improved method and apparatus for balancing distributed applications within a client/server network, such as a cable television network, is disclosed. In one aspect of the invention, a method of balancing the load of distributed application client portions (DACPs) across various server portions (DASPs) and server machines is disclosed. Statistics are maintained by one or more software processes with respect to the available resources of the servers and their loading; new process threads and/or distributed application server portions are allocated across the servers to maintain optimal system performance as client device loading increases or changes. In another aspect of the invention, a novel object-oriented distributed application software architecture employing both vertical and horizontal partitions and "mutable" (i.e., transportable) objects is disclosed. The mutable objects may reside on either the server or client portions of the distributed application while maintaining at least one network partition. A runtime environment adapted for the operation of the foregoing object-oriented distributed application, including an efficient message protocol useful for interprocess communication, is also disclosed. Methods for downloading the DACP from the servers, and scaling the DACP at download based on client device configuration, are further disclosed.

3 Claims, 28 Drawing Sheets



Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L3: Entry 3 of 5 File: USPT Apr 27, 1999

DOCUMENT-IDENTIFIER: US 5898830 A
TITLE: Firewall providing enhanced network security and user transparency

Abstract Text (1):
The present invention, generally speaking, provides a firewall that achieves maximum network security and maximum user convenience. The firewall employs "envoys" that exhibit the security robustness of prior-art proxies and the transparency and ease-of-use of prior-art packet filters, combining the best of both worlds. No traffic can pass through the firewall unless the firewall has established an envoy for that traffic. Both connection-oriented (e.g., TCP) and connectionless (e.g., UDP-based) services may be handled using envoys. Establishment of an envoy may be subjected to a myriad of tests to "qualify" the user, the requested communication, or both. Therefore, a high level of security may be achieved. The usual added burden of prior-art proxy systems is avoided in such a way as to achieve fall transparency-the user can use standard applications and need not even know of the existence of the firewall. To achieve full transparency, the firewall is configured as two or more sets of virtual hosts. The firewall is, therefore, "multi-homed," each home being independently configurable. One set of hosts responds to addresses on a first network interface of the firewall. Another set of hosts responds to addresses on a second network interface of the firewall. In one aspect, programmable transparency is achieved by establishing DNS mappings between remote hosts to be accessed through one of the network interfaces and respective virtual hosts on that interface. In another aspect, automatic transparency may be achieved using code for dynamically mapping remote hosts to virtual hosts in accordance with a technique referred to herein as dynamic DNS, or DDNS.

CLAIMS:

9. The method of claim 4, comprising the further steps of, for at least one of the firewalls:

providing multiple physical computers, each configured as a plurality of virtual hosts, a first virtual host on one of said physical machines being identically configured as a second virtual host on another of said physical machines;

wherein said mapping from a name of the second computer to a network address of one of the virtual hosts of the firewall is made dynamically to one of said first virtual host and said second virtual host depending on availability of said one physical machine and said another physical machine.

Previous Doc Next Doc Go to Doc#

h e b b g e e e f e c e g

US005898830A

United States Patent [19]

Wesinger, Jr. et al.

[11] Patent Number: 5,898,830

[45] Date of Patent: Apr. 27, 1999

[54] **FIREWALL PROVIDING ENHANCED NETWORK SECURITY AND USER TRANSPARENCY**

[75] Inventors: **Ralph E. Wesinger, Jr.**, San Jose; **Christopher D. Coley**, Morgan Hill, both of Calif.

[73] Assignee: **Network Engineering Software**, San Jose, Calif.

[21] Appl. No.: **08/733,361**

[22] Filed: **Oct. 17, 1996**

[51] **Int. Cl.**[6] **G06F 1/00**

[52] **U.S. Cl.** **395/187.01**; 395/200.55; 395/200.57

[58] **Field of Search** 395/186, 187.01, 395/188.01, 200.3, 200.55, 200.68, 200.57; 380/3, 4, 21, 23, 25; 340/825.3

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,713,753	12/1987	Boebert et al.	364/200
4,799,153	1/1989	Hann et al.	380/25
4,799,156	1/1989	Shavit et al.	364/401
5,191,611	3/1993	Lang	380/25
5,241,594	8/1993	Kung	380/4
5,416,842	5/1995	Aziz	380/30

(List continued on next page.)

OTHER PUBLICATIONS

Kiuchi et al., "C-HTTP The Development of a Secure, Closed HTTP Based Network on the Internet", PRoceedings of SNDSS, IEEE, pp. 64–75, Jun. 1996.

Neuman, "Proxy Based Authorization and Accounting for Distributed Systems", IEEE, pp. 283–291, 1993.

Network Firewalls; *IEEE Communications Magazine*; (Ballovin et al.) pp. 50–57; Sep., 1994.

The MITRE Security Perimeter; *IEEE Communications Magazine*; (Goldberg); pp. 212–218; 1994.

IpAccess—An Internet Service Access System for Firewall Installations; *IEEE Communications Magazine*; (Stempel); pp. 31–41; 1995.

Remote Control of Diverse Network Elements Using SNMP; *IEEE Communications Magazine*; (Aicklen et al.); pp. 673–667; 1995.

Firewall's Information is Money!, *Scientific Management Corporation.*

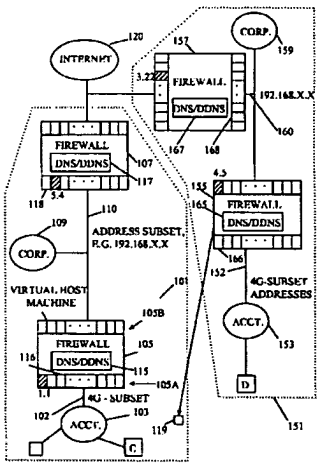
Primary Examiner—Joseph Palys

Attorney, Agent, or Firm—McDonnell Boehnen Hulbert & Berghoff

[57] **ABSTRACT**

The present invention, generally speaking, provides a firewall that achieves maximum network security and maximum user convenience. The firewall employs "envoys" that exhibit the security robustness of prior-art proxies and the transparency and ease-of-use of prior-art packet filters, combining the best of both worlds. No traffic can pass through the firewall unless the firewall has established an envoy for that traffic. Both connection-oriented (e.g., TCP) and connectionless (e.g., UDP-based) services may be handled using envoys. Establishment of an envoy may be subjected to a myriad of tests to "qualify" the user, the requested communication, or both. Therefore, a high level of security may be achieved. The usual added burden of prior-art proxy systems is avoided in such a way as to achieve fall transparency-the user can use standard applications and need not even know of the existence of the firewall. To achieve full transparency, the firewall is configured as two or more sets of virtual hosts. The firewall is, therefore, "multi-homed," each home being independently configurable. One set of hosts responds to addresses on a first network interface of the firewall. Another set of hosts responds to addresses on a second network interface of the firewall. In one aspect, programmable transparency is achieved by establishing DNS mappings between remote hosts to be accessed through one of the network interfaces and respective virtual hosts on that interface. In another aspect, automatic transparency may be achieved using code for dynamically mapping remote hosts to virtual hosts in accordance with a technique referred to herein as dynamic DNS, or DDNS.

21 Claims, 9 Drawing Sheets



Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L3: Entry 4 of 5 File: USPT Mar 2, 1999

DOCUMENT-IDENTIFIER: US 5878212 A
TITLE: System for updating mapping or virtual host names to layer-3 address when multimedia server changes its usage state to busy or not busy

Abstract Text (1):
A name mapper, name servers, and multimedia servers are connected to a multimedia manager. Each client has the name of a multimedia server, i.e., a virtual host name, from which it can obtain multimedia service. The name server stores associations of server host names to layer-3 addresses. When a client initiates a multimedia session, it requests the layer-3 address of the server that corresponds to its server's name. The name server sends the layer-3 address of the one of the multimedia servers that is currently designated as corresponding to that name. The multimedia client stores the name-to-layer-3 address binding in it's cache. The multimedia client then establishes communications with the multimedia server at that layer-3 address and clears its cache. The dynamic name-to-layer-3 address binding in the name server is managed by the name mapper, which may be collocated with the multimedia manager or may be located on a separate server. The multimedia server manager collects real-time status information so that it knows the availability of the multimedia servers in the network. If a multimedia server, whose layer-3 address is presently mapped to from a virtual host name, becomes unable to serve additional clients, the multimedia server manager sends a message to the name mapper to modify the name to layer-3 address binding. The modification specifies an available server's layer-3 address to be bound in place of that of the server that became unable to serve additional clients.

Brief Summary Text (19):
If a multimedia server, whose layer-3 address is presently associated with a virtual host name, becomes unable to serve additional clients, the multimedia server manager sends a message to the name mapper to modify the dynamic name to layer-3 address binding. The modification specifies a new binding, by designating in the current server binding table an available server's layer-3 address in place of the server that became unable to serve additional clients. The name mapper then automatically changes the binding in the name servers.

Previous Doc Next Doc Go to Doc#

US005878212A

United States Patent [19]

Civanlar et al.

[11] Patent Number: 5,878,212

[45] Date of Patent: Mar. 2, 1999

[54] **SYSTEM FOR UPDATING MAPPING OR VIRTUAL HOST NAMES TO LAYER-3 ADDRESS WHEN MULTIMEDIA SERVER CHANGES ITS USAGE STATE TO BUSY OR NOT BUSY**

[75] Inventors: **Seyhan Civanlar**, Middletown Township, Monmouth County; **Vikram R. Saksena**, Freehold, both of N.J.

[73] Assignee: **AT&T Corp.**, Middletown, N.J.

[21] Appl. No.: **509,308**

[22] Filed: **Jul. 31, 1995**

[51] **Int. Cl.**[6] **G06F 13/00**

[52] **U.S. Cl.** **395/200.33**; 395/200.49

[58] **Field of Search** 395/200.33, 200.49, 395/200.5, 200.51, 200.53, 200.54, 200.56, 200.57, 200.58, 200.59, 200.75, 182.02, 684; 370/392

[56] **References Cited**

U.S. PATENT DOCUMENTS

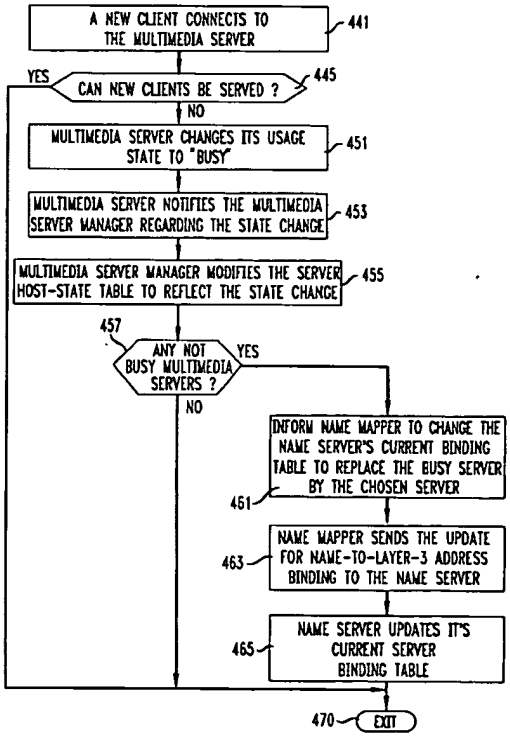
4,800,488	1/1989	Agrawal et al.	395/200.55
5,025,491	6/1991	Tsuchiya et al.	370/255
5,227,778	7/1993	Vacon et al.	370/445
5,341,477	8/1994	Pitkin et al.	395/200.56
5,475,819	12/1995	Miller et al.	395/200.33
5,483,652	1/1996	Sudama et al.	707/10
5,592,611	1/1997	Midgely et al.	395/182.02
5,594,921	1/1997	Pettus	395/831
5,608,865	3/1997	Midgely et al.	395/180

Primary Examiner—Moustafa M. Meky

[57] **ABSTRACT**

A name mapper, name servers, and multimedia servers are connected to a multimedia manager. Each client has the name of a multimedia server, i.e., a virtual host name, from which it can obtain multimedia service. The name server stores associations of server host names to layer-3 addresses. When a client initiates a multimedia session, it requests the layer-3 address of the server that corresponds to its server's name. The name server sends the layer-3 address of the one of the multimedia servers that is currently designated as corresponding to that name. The multimedia client stores the name-to-layer-3 address binding in it's cache. The multimedia client then establishes communications with the multimedia server at that layer-3 address and clears its cache. The dynamic name-to-layer-3 address binding in the name server is managed by the name mapper, which may be collocated with the multimedia manager or may be located on a separate server. The multimedia server manager collects real-time status information so that it knows the availability of the multimedia servers in the network. If a multimedia server, whose layer-3 address is presently mapped to from a virtual host name, becomes unable to serve additional clients, the multimedia server manager sends a message to the name mapper to modify the name to layer-3 address binding. The modification specifies an available server's layer-3 address to be bound in place of that of the server that became unable to serve additional clients.

20 Claims, 5 Drawing Sheets



Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L3: Entry 5 of 5 File: USPT Apr 1, 1997

DOCUMENT-IDENTIFIER: US 5617540 A
TITLE: System for binding host name of servers and address of available server in cache within client and for clearing cache prior to client establishes connection

Abstract Text (1):
A name mapper, name servers, and multimedia servers are connected to a multimedia manager. Each client has the name of a multimedia server, i.e., a virtual host name, from which it can obtain multimedia service. The name server stores associations of server host names to layer-3 addresses. When a client initiates a multimedia session, it requests the layer-3 address of the server that corresponds to its server's name. The name server sends the layer-3 address of the one of the multimedia servers that is currently designated as corresponding to that name. The multimedia client stores the name-to-layer-3 address binding in it's cache. The multimedia client then establishes communications with the multimedia server at that layer-3 address and clears its cache. The dynamic name-to-layer-3 address binding in the name server is managed by the name mapper, which may be collocated with the multimedia manager or may be located on a separate server. The multimedia server manager collects real-time status information so that it knows the availability of the multimedia servers in the network. If a multimedia server, whose layer-3 address is presently mapped to from a virtual host name, becomes unable to serve additional clients, the multimedia server manager sends a message to the name mapper to modify the name to layer-3 address binding. The modification specifies an available server's layer-3 address to be bound in place of that of the server that became unable to serve additional clients.

Brief Summary Text (19):
If a multimedia server, whose layer-3 address is presently associated with a virtual host name, becomes unable to serve additional clients, the multimedia server manager sends a message to the name mapper to modify the dynamic name to layer-3 address binding. The modification specifies a new binding, by designating in the current server binding table an available server's layer-3 address in place of the server that became unable to serve additional clients. The name mapper then automatically changes the binding in the name servers.

Previous Doc Next Doc Go to Doc#

US005617540A

United States Patent [19]

Civanlar et al.

[11] Patent Number: 5,617,540

[45] Date f Patent: Apr. 1, 1997

[54] **SYSTEM FOR BINDING HOST NAME OF SERVERS AND ADDRESS OF AVAILABLE SERVER IN CACHE WITHIN CLIENT AND FOR CLEARING CACHE PRIOR TO CLIENT ESTABLISHES CONNECTION**

[75] Inventors: **Seyhan Civanlar**, Middletown Township, Monmouth County; **Vikram R. Saksena**, Freehold, both of N.J.

[73] Assignee: **AT&T**, Middletown, N.J.

[21] Appl. No.: **509,307**

[22] Filed: **Jul. 31, 1995**

[51] **Int. Cl.**[6] **G06F 13/00**

[52] **U.S. Cl.** **395/200.11**

[58] **Field of Search** 395/823, 824, 395/829, 846, 200.02, 200.06, 200.09, 200.11, 200.12

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,025,491	6/1991	Tsuchiya et al.	340/825.52
5,136,716	8/1992	Harvey et al.	395/800
5,287,103	2/1994	Kaspryzk et al.	340/825.52
5,454,078	9/1995	Heimsoth et al.	395/200.1
5,463,735	10/1995	Pascucci et al.	395/200.1
5,526,489	6/1996	Nilakantan et al.	395/200.02

OTHER PUBLICATIONS

"Mosaic and the World–Wide Web", Vetter et al, IEEE Oct. 1994, pp. 49–51.

"Multiple Protocol Network Integration: A Case Study in Internetworking with IP/IPX", IEEE 92, pp. 560–567.

"Understanding Data Communication & Networks", Shay, PWS Publishing, ©1994, pp. 417–421; 440–449.

"OSI –An Appropriate Basis for Group Communications ?" Jakobs, IEEE Sep. 1989, pp. 346–350.

"IP Addressing & Routing in a Local Wireless Network", Cohen et al, IEEE 92, pp. 626–632.

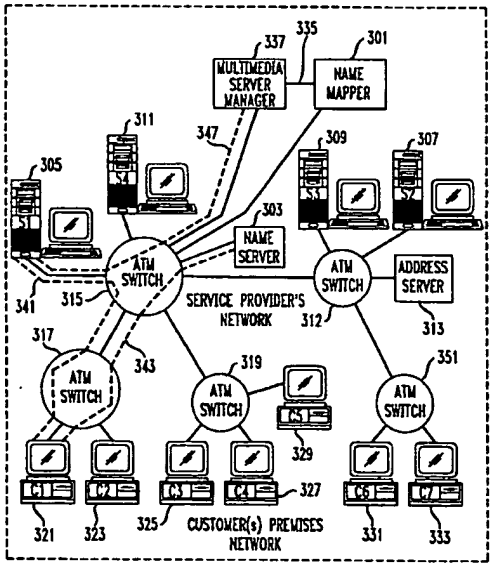
"Ubiquitous Mobile Host Internetworking" Johnson, IEEE 93, pp. 85–90.

Primary Examiner—Thomas C. Lee
Assistant Examiner—Moustafa Mohamed Meky
Attorney, Agent, or Firm—Stephen M. Gurey

[57] **ABSTRACT**

A name mapper, name servers, and multimedia servers are connected to a multimedia manager. Each client has the name of a multimedia server, i.e., a virtual host name, from which it can obtain multimedia service. The name server stores associations of server host names to layer-3 addresses. When a client initiates a multimedia session, it requests the layer-3 address of the server that corresponds to its server's name. The name server sends the layer-3 address of the one of the multimedia servers that is currently designated as corresponding to that name. The multimedia client stores the name-to-layer-3 address binding in it's cache. The multimedia client then establishes communications with the multimedia server at that layer-3 address and clears its cache. The dynamic name-to-layer-3 address binding in the name server is managed by the name mapper, which may be collocated with the multimedia manager or may be located on a separate server. The multimedia server manager collects real-time status information so that it knows the availability of the multimedia servers in the network. If a multimedia server, whose layer-3 address is presently mapped to from a virtual host name, becomes unable to serve additional clients, the multimedia server manager sends a message to the name mapper to modify the name to layer-3 address binding. The modification specifies an available server's layer-3 address to be bound in place of that of the server that became unable to serve additional clients.

4 Claims, 5 Drawing Sheets



US005898830A

United States Patent [19]

Wesinger, Jr. et al.

[11] Patent Number: 5,898,830

[45] Date of Patent: Apr. 27, 1999

[54] **FIREWALL PROVIDING ENHANCED NETWORK SECURITY AND USER TRANSPARENCY**

[75] Inventors: **Ralph E. Wesinger, Jr.**, San Jose; **Christopher D. Coley**, Morgan Hill, both of Calif.

[73] Assignee: **Network Engineering Software**, San Jose, Calif.

[21] Appl. No.: **08/733,361**

[22] Filed: **Oct. 17, 1996**

[51] **Int. Cl.**[6] **G06F 1/00**

[52] **U.S. Cl.** **395/187.01**; 395/200.55; 395/200.57

[58] **Field of Search** 395/186, 187.01, 395/188.01, 200.3, 200.55, 200.68, 200.57; 380/3, 4, 21, 23, 25; 340/825.3

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,713,753	12/1987	Boebert et al.	364/200
4,799,153	1/1989	Hann et al.	380/25
4,799,156	1/1989	Shavit et al.	364/401
5,191,611	3/1993	Lang	380/25
5,241,594	8/1993	Kung	380/4
5,416,842	5/1995	Aziz	380/30

(List continued on next page.)

OTHER PUBLICATIONS

Kiuchi et al., "C-HTTP The Development of a Secure, Closed HTTP Based Network on the Internet", PRoceedings of SNDSS, IEEE, pp. 64–75, Jun. 1996.

Neuman, "Proxy Based Authorization and Accounting for Distributed Systems", IEEE, pp. 283–291, 1993.

Network Firewalls; *IEEE Communications Magazine*; (Ballovin et al.) pp. 50–57; Sep., 1994.

The MITRE Security Perimeter; *IEEE Communications Magazine*; (Goldberg); pp. 212–218; 1994.

IpAccess—An Internet Service Access System for Firewall Installations; *IEEE Communications Magazine*; (Stempel); pp. 31–41; 1995.

Remote Control of Diverse Network Elements Using SNMP; *IEEE Communications Magazine*; (Aicklen et al.); pp. 673–667; 1995.

Firewall's Information is Money!, *Scientific Management Corporation.*

Primary Examiner—Joseph Palys

Attorney, Agent, or Firm—McDonnell Boehnen Hulbert & Berghoff

[57] **ABSTRACT**

The present invention, generally speaking, provides a firewall that achieves maximum network security and maximum user convenience. The firewall employs "envoys" that exhibit the security robustness of prior-art proxies and the transparency and ease-of-use of prior-art packet filters, combining the best of both worlds. No traffic can pass through the firewall unless the firewall has established an envoy for that traffic. Both connection-oriented (e.g., TCP) and connectionless (e.g., UDP-based) services may be handled using envoys. Establishment of an envoy may be subjected to a myriad of tests to "qualify" the user, the requested communication, or both. Therefore, a high level of security may be achieved. The usual added burden of prior-art proxy systems is avoided in such a way as to achieve fall transparency-the user can use standard applications and need not even know of the existence of the firewall. To achieve full transparency, the firewall is configured as two or more sets of virtual hosts. The firewall is, therefore, "multi-homed," each home being independently configurable. One set of hosts responds to addresses on a first network interface of the firewall. Another set of hosts responds to addresses on a second network interface of the firewall. In one aspect, programmable transparency is achieved by establishing DNS mappings between remote hosts to be accessed through one of the network interfaces and respective virtual hosts on that interface. In another aspect, automatic transparency may be achieved using code for dynamically mapping remote hosts to virtual hosts in accordance with a technique referred to herein as dynamic DNS, or DDNS.

21 Claims, 9 Drawing Sheets